(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 4 481 744 A1**

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
25.12.2024 Bulletin 2024/52

(21) Application number: **24178167.3**

(22) Date of filing: **27.05.2024**

(51) International Patent Classification (IPC):
G11C 29/00 *(2006.01)* ***G11C 5/02*** *(2006.01)*
G11C 29/44 *(2006.01)* *G11C 7/10* *(2006.01)*
G11C 17/18 *(2006.01)* *G11C 11/54* *(2006.01)*

(52) Cooperative Patent Classification (CPC):
G11C 29/4401; G11C 5/025; G11C 29/785; G11C 29/812; G11C 29/86; G11C 7/1006; G11C 11/54; G11C 17/18; G11C 2029/4402

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC ME MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA
Designated Validation States:
GE KH MA MD TN

(30) Priority: **20.06.2023 US 202318337751**

(71) Applicant: **STMicroelectronics International N.V.**
1228 Plan-les-Ouates, Geneva (CH)

(72) Inventors:
- **KARSANBHAI PATEL, Urmishkumar**
  396521 Chikhali, Gujarat (IN)
- **SYED, Danish Hasan**
  110025 New Delhi (IN)
- **SINGH, Prateek**
  110051 Delhi (IN)

(74) Representative: **Buzzi, Notaro & Antonielli d'Oulx S.p.A.**
Corso Vittorio Emanuele II, 6
10123 Torino (IT)

(54) COMMUNICATION LOGIC TO ENHANCE AREA EFFECTIVENESS FOR MEMORY REPAIR MECHANISM

(57) According to an embodiment, a method for testing and repairing local memory (108) in a hardware accelerator (100) from a one-time programmable memory, OTP (502) is provided. The method includes asserting a grant signal (804), a loading of a first repair data for a sub-set of the local memory (108) associated with a main-controller (706a) from a first partition (602) of the OTP memory, communicating a status signal (806) after completion of the loading indicating a completion of the loading, and de-asserting the grant signal (804) in response to receiving the status signal (806).

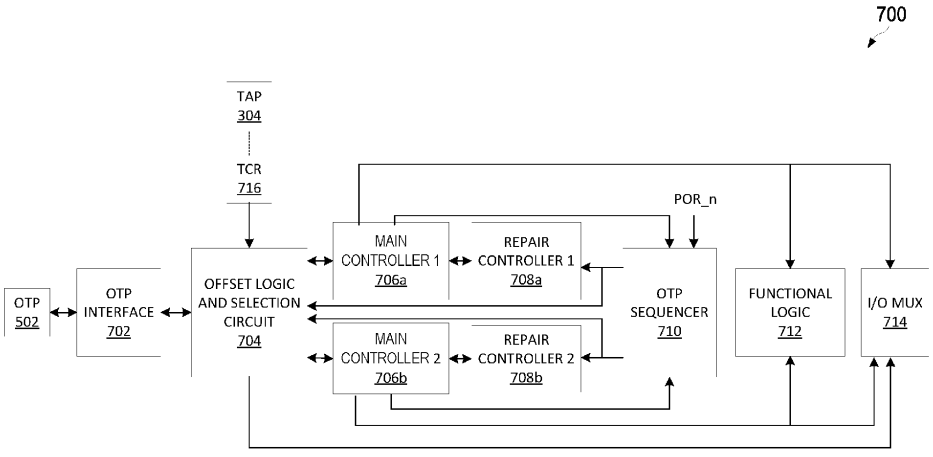

FIG. 7

Processed by Luminess, 75001 PARIS (FR)

Description

TECHNICAL FIELD

[0001] The present disclosure generally relates to design for testing (DFT) and, in particular embodiments, to an improved system architecture for testing and repairing memory.

BACKGROUND

[0002] Hardware accelerators, such as Graphics Processing Units (GPUs), Field Programmable Gate Arrays (FPGAs), and Application-Specific Integrated Circuits (ASICs) are specialized hardware devices designed to perform specific tasks more efficiently than traditional general-purpose processors. The use of hardware accelerators in machine learning and artificial intelligence has become increasingly important as the size of datasets and complexity of models have increased, which has led to an increase in both memory count and memory spread across the system-on-chip (SoC).

[0003] In the context of AI and machine learning, hardware accelerators can speed up the computation-intensive tasks involved in the training and inference of machine learning models. By offloading computation to specialized hardware, machine learning algorithms can be trained and deployed more quickly and efficiently, leading to better performance and faster time to market for AI-based products and services.

[0004] Generally, hardware accelerators require significant amounts of memory space to operate effectively and large amounts of data are stored to, for example, train datasets, model parameters, and compute intermediate results. Thus, the memory in hardware accelerators is becoming increasingly complex, with more memory channels and higher memory bandwidth to support the increasing demands of machine learning and AI workloads. Additionally, as the size of the workloads continues to grow, the amount of memory required to support them also increases. Traditionally, a hierarchical approach has been used to test and repair memory, simplifying the memory repair process by abstracting the details of the underlying memory subsystem and providing a standardized interface for error detection and correction. In the hierarchical approach, the memory subsystem is divided into multiple levels of abstraction, each with its own set of tests and wrappers. The top-level wrapper interfaces with the external environment and generates test patterns for the lower-level wrappers. The lower-level wrappers verify the functionality and performance of the memory at their respective levels of abstraction and provide feedback to the top-level wrapper.

[0005] Conventionally, the loading of the instructions through a test access point (TAP) and communicating with a one-time programmable (OTP) memory is done through the hierarchical approach's top main-controller/-sub-controller system architecture. Due to synthesis constraints and many processing cores, multiple controllers are used in artificial intelligence and machine learning applications, necessitating a dedicated OTP memory for each system. However, each system does not necessarily use the entire OTP memory space to store repair data, which results in memory wastage.

[0006] A memory test processor calculates a repair signature based on the memory fault location to repair memory using the hierarchical approach. The repair signature is stored in the OTP memory during the manufacturing test. In the field, during the power-up sequence, the repair signature is read from the OTP memory and loaded to the memory repair signature registers.

[0007] Thus, an improved system architecture to test and repair memory is desirable.

SUMMARY

[0008] Technical advantages are generally achieved by embodiments of this disclosure which describe an improved system architecture for testing and repairing memory. A first aspect relates to a hardware accelerator including a system for testing and repairing local memory of the hardware accelerator. The system includes a one-time programmable (OTP) memory, a memory repair control circuit (e.g., an OTP sequencer circuit or a test control register, TCR), a first main-controller, a second main-controller, and an offset addition and selection logic circuit. The one-time programmable memory is configured to sequentially store first repair data in a first partition and second repair data in a second partition for, respectively, a first subset of the local memory and a second subset of the local memory. The memory repair control circuit is configured to sequentially assert a first grant signal and a second grant signal. The first main-controller is coupled to the first subset of the local memory. The first main-controller is configured to first load the first repair data for the first subset of the local memory from the first partition in response to the first grant signal being asserted. The offset addition and selection logic circuit is configured to add an offset to an address used by the second main-controller to access the second partition. The second main-controller is coupled to the second subset of the local memory. The second main-controller is configured to second load the second repair data for the second subset of the local memory from the second partition using the address to access the second partition, in response to the second grant signal being asserted.

[0009] A second aspect relates to a method for testing and repairing local memory in a hardware accelerator from a one-time programmable (OTP) memory. The method includes asserting, by a memory repair control circuit, a first grant signal; a first loading, by a first main-controller coupled to a first subset of the local memory, of first repair data for the first subset of the local memory from a first partition of the OTP memory in response to the

first grant signal being asserted; asserting, by the memory repair control circuit, a second grant signal; adding, by an offset addition and selection logic circuit, an offset to an address used by a second main-controller coupled to a second subset of the local memory to access a second partition of the OTP memory comprising second repair data, wherein the first repair data and the second repair data are sequentially stored in the OTP memory; a second loading, by the second main-controller, of the second repair data for the second subset of the local memory from the second partition of the OTP memory using the address to access the second partition in response to the second grant signal being asserted. Embodiments can be implemented in hardware, software, or any combination thereof.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] For a more complete understanding of the present disclosure and the advantages thereof, reference is now made to the following descriptions taken in conjunction with the accompanying drawings, in which:

Figure 1 is a block diagram of an embodiment hardware accelerator;
Figures 2, 3, and 4, respectively, are a simplified block diagram of a conventional design for testing (DFT) sub-system, system, and multi-system, based on the hierarchical approach;
Figure 5 is a simplified block diagram of an embodiment multi-system;
Figure 6 is a partitioning of an embodiment OTP memory;
Figure 7 is a block diagram of an embodiment design for system;
Figures 8 and 9, respectively, are an embodiment timing diagram and a corresponding method where the system in Figure 7 is operating in functional mode; and
Figures 10 and 11, respectively, are an embodiment timing diagram and a corresponding method where the system in Figure 7 is operating in test mode.

DETAILED DESCRIPTION OF ILLUSTRATIVE EMBODIMENTS

[0011] This disclosure provides many applicable inventive concepts that can be embodied in a wide variety of specific contexts. The particular embodiments are merely illustrative of specific configurations and do not limit the scope of the claimed embodiments. Features from different embodiments may be combined to form further embodiments unless noted otherwise.

[0012] Variations or modifications described in one of the embodiments may also apply to others. Further, various changes, substitutions, and alterations can be made herein without departing from the spirit and scope of this disclosure as defined by the appended claims.

[0013] While the inventive aspects are described primarily in the context of a hardware accelerator used in artificial intelligence or machine learning applications, it should also be appreciated that these inventive aspects may also apply to other applications that benefit from efficient, compact, and optimized design for test schemes.

[0014] Figure 1 illustrates a block diagram of an embodiment hardware accelerator 100. As shown, hardware accelerator 100 includes multiple interconnected cores 102 (i.e., logic blocks), which provides a compact, efficient, flexible, and scalable solution, enabling the integration of diverse functionality into a single chip. In embodiments, each core 102 includes a vast number of gates 104, a processing unit (PU) 106, and local memory (LM) 108, which may (or may not) be arranged as shown. Further, hardware accelerator 100 may include additional components not shown. It is noted that the number of cores 102, gates 104, processing unit 106, and local memory 108 are non-limiting.

[0015] In embodiments, hardware accelerator 100 is integrated into an artificial intelligence or machine learning system to process large computations in a distributed manner. The combination of parallel architecture, numerous processing units, distributed on-chip memory, and optimized interconnectivity allows the hardware accelerator 100 to achieve high throughput, efficiently process massive amounts of data, and perform quick, complex computations.

[0016] Each core 102 may include many gates 104 allowing for intricate designs and integrating numerous functional units and features. In embodiments, hardware accelerator 100 may include cores 102. Cores 102 may represent different functional units or subsystems, such as central processing units (CPUs), graphical processing units (GPUs), peripherals, or custom IP blocks.

[0017] Moreover, hardware accelerator 100 can include a massive number of interconnected processing units 106 that operate in parallel to, for example, simultaneously perform computations on one or more problem sets. Processing units 106 typically include CPU cores, GPUs, custom accelerators, or a combination thereof, that can be highly specialized and optimized to perform specific tasks and execute high-throughput operations.

[0018] Hardware accelerator 100 includes many distributed on-chip, local memory 108 to facilitate low latency operations on localized data. Generally, the local memory 108 is closely located to the processing unit 106 to minimize latency and enable quick data sharing among the processing unit 106. Local memory 108 can be of different types, such as RAM (Random Access Memory), ROM (Read-Only Memory), or other specialized memory technologies.

[0019] Local memory 108 provides storage and data access for the specific functions performed by the respective core 102. Integrating the local memory 108 within each core 102 advantageously allows for faster and more efficient data access as the local memory 108 is

closer to the processing unit 106. Additionally, the integration enables the customization and optimization of memory resources based on the specific requirements of each core 102. Different cores 102 may have varying memory capacities, types, or architectures tailored to their needs. Furthermore, having a core 102 with its own dedicated local memory 108 facilitates modularity and scalability in the system design. Each core 102 can operate independently, accessing and managing its associated local memory 108 without interfering with others. This modularity allows for easier development, testing, and maintenance of individual components within the hardware accelerator 100.

[0020] The processing units 106 and local memory 108 (and optionally, other components of the core 102) are typically interconnected using a highly optimized network called network-on-chip 110 (NoC). The network-on-chip 110 provides efficient communication channels and high-bandwidth data transfer among the processing units 106, local memory 108, gates 104, and other components of the hardware accelerator 100.

[0021] As noted, traditionally, a hierarchical approach has been used to test and repair memory by abstracting away the details of the underlying memory sub-system and providing, for example, a standardized interface for error detection and correction. To achieve reasonable test times, the full set of cores 102 of the hardware accelerator 100 are divided into smaller sub-sets and multiple levels of hierarchy. Figures 2, 3, and 4, respectively, illustrate a simplified block diagram of a conventional design for testing (DFT) sub-system 200, system 300, and multi-system 400, based on the hierarchical approach. Thus, the sub-system 200 is used to test lower-level blocks of hierarchy, system 300 is used to test multiple sub-systems 200, and multi-system 400 is used to test multiple systems 300.

[0022] In embodiments, sub-system 200 may include multiple cores 102. As shown in Figure 2, a sub-system 200 includes a sub-controller 202, multiple memory test processors 204, and multiple local memory 108, which may (or may not) be arranged as shown. Sub-system 200 may include additional components not shown. The number of components associated with the sub-system 200 is non-limiting.

[0023] Each local memory 108 is wrapped within an individual wrapper 206. Each wrapper 206 is associated and coupled to a memory test processor 204. In embodiments, each wrapper 206 includes multiple local memory 108. In some embodiments, multiple wrappers 206 are associated and coupled to a memory test processor 204. In embodiments, the memory test processor 204 is the processing unit 106 of the core 102 associated with the local memory 108. Each memory test processor 204 is configured to test the local memory 108 associated with the wrapper 206. A sub-controller 202 is added to the core 102 and coupled to each memory test processor 204 within the sub-system 200.

[0024] In embodiments, system 300 may include multiple sub-systems 200 coupled to a main-controller 302, added to the hardware accelerator 100. System 300 may (or may not) be arranged as shown and may include additional components not shown. In embodiments, the main-controller 302 is coupled to the test access point (TAP) 304 and the one-time programmable (OTP) memory 306. Main-controller 302 is configured to allow the sub-systems 200 to communicate with a common TAP 304. The number of components associated with the system 300 is non-limiting.

[0025] TAP 304, or boundary scan, tests and diagnoses digital circuits, including memory subsystems. It generally involves adding dedicated test logic to a circuit that can be used to access and control individual device pins, enabling the testing of internal signals and the detection of faults in the circuit. In memory tests and repair, TAP 304 can access individual memory cells and perform various diagnostic tests, such as read/write, bit-wise comparison, and pattern testing. It can also detect and isolate faults in the memory subsystem.

[0026] OTP memory 306, is a type of non-volatile memory that can be programmed only once during manufacturing and retains its value indefinitely. In the context of memory testing and repair, OTP memory 306 can store configuration and repair data used during the memory testing and repair process.

[0027] For example, a memory test and repair system may use OTP memory 306 to store a repair map that identifies faulty memory cells and their replacement values. During testing, the system can read the repair map from the OTP memory 306 and use it to replace the faulty memory cells with the correct values stored in the OTP memory 306. OTP memory 306 can also store configuration data, such as test patterns, timing parameters, and other settings used during testing and repair. Thus, TAP 304 enables the testing and diagnosis of the memory subsystem, while OTP memory 306 provides a reliable and non-volatile storage solution for the configuration and repair of data.

[0028] For example, in system 300 where the number of sub-systems 200 equals one, the sub-system 200 is coupled to an OTP memory 306 of 256 words through a main-controller 302. In this example, the repair data for the first local memory 108 of the sub-system 200 is stored in the first word of the OTP memory 306, the repair data for the second local memory 108 of the sub-system 200 is stored in the second word of the OTP memory 306, and so forth. In this example, the number of local memory 108 of the sub-system 200 is less than 256 (e.g., thirty-two). As such OTP memory 306 includes a significant amount of memory space that is wasted due to inefficient communication between OTP memory 306 and main controller 302.

[0029] Typically, in a system-on-chip not used in artificial learning or machine learning applications, during manufacturing (i.e., production), instructions are loaded from the TAP 304 into the memory test processor 204 to test and repair the local memory 108. During the repair

process, repair signatures are calculated and stored in the OTP memory 306. The loading of instructions from the TAP 304 and communication with the OTP memory 306 are done using the sub-system 200 and system 300 DFT architectures. With the increased number of cores 102 in a hardware accelerator 100 used in artificial intelligence and machine learning applications, the design for test architecture has expanded to include a higher-level block above the system 300, known as the multi-system 400.

[0030] In embodiments, multi-system 400 may include multiple systems 300, where each system 300 is coupled to a respective, dedicated OTP memory 402. Multi-system 400 may (or may not) be arranged as shown and may include additional components not shown.

[0031] Disadvantageously, each system 300 coupled to the respective OTP memory 402 does not utilize the entire memory space of the respective OTP memory 402 for data storage repair. Thus, the usage of a dedicated OTP memory 402 for each system 300 results in wasted and unused space in each OTP memory 402, which, when multiplied by the number of systems 300 in the multi-system 400, results in a large amount of wasted memory space that can be otherwise used for other purposes.

[0032] For example, in a multi-system 400 where the number of systems 300 equals two, each system 300 is coupled to a respective OTP memory 402 of 256 words through a respective main-controller 302. In this example, the repair data for the set of local memory 108 of the first system 300 is stored in the first OTP memory 402 and the repair data for the set of local memory 108 of the second system 300 is stored in the second OTP memory 402.

[0033] Further, the repair data for the first local memory 108 in the set of local memory 108 of the first system 300 is stored in the first word of the first OTP memory 402, the repair data for the second local memory 108 in the set of local memory 108 of the first system 300 is stored in the second word of the first OTP memory 402, and so forth.

[0034] Moreover, the repair data for the first local memory 108 in the set of local memory 108 of the second system 300 is stored in the first word of the second OTP memory 402, the repair data for the second local memory 108 in the set of local memory 108 of the second system 300 is stored in the second word of the second OTP memory 402, and so forth.

[0035] In this example, the number of local memory 108 is less than 256. For example, the quantity of local memory 108 in the set of local memory 108 of the first system 300 is thirty-two and the quantity of local memory 108 in the set of local memory 108 of the second system 300 is twenty. Thus, in the multi-system 400, each OTP memory 402 includes a significant amount of memory space that is wasted due to inefficient communication between the OTP memory 402 and the main controller 302. Further, it is observable that as the size of the local memory 108 is increased in, for example, artificial intelligence and machine learning systems, the use of a dedicated OTP memory 402 for each system 300 results in significant waste in memory space, resulting in a poor memory utilization system.

[0036] To address this issue, some design for test architectures include different controllers (e.g., main-controller 302) for repairable and non-repairable memory. Disadvantageously, this solution is limited to only two controllers. A core 102 may contain repairable and non-repairable memory, resulting in complicated timing closure and increased spatial memory streaming (SMS) logic per system block.

[0037] Embodiments of this disclosure provide an improved design for test architecture for testing and repairing memory (e.g., local memory 108) that addresses the deficiencies in the existing solutions. In embodiments, a single OTP memory is coupled to the many systems in the multi-system to improve the efficient usage of the OTP memory space in a hardware accelerator. A communication logic circuit is coupled between the single OTP memory and the many systems, which allows each main-controller 302 to communicate with the single OTP memory. Aspects of this disclosure provide a design for test methodology for a system-on-chip used in artificial intelligence or machine learning technology, such as automotive applications. Embodiments of this disclosure similarly apply to production (i.e., manufacturing) and functional (i.e., boot mode) testing. These and further details are discussed in greater detail below.

[0038] Figure 5 illustrates a simplified block diagram of an embodiment multi-system 500, which may (or may not) be arranged as shown. In contrast to the multi-system 400 in Figure 4, the multi-system 500 in Figure 5 includes a single OTP memory 502 coupled to each main-controller 302 of each system 300 through a communication logic circuit 504. Each main-controller 302 is coupled to multiple sub-systems 200. Multi-system 500 may include additional components not shown, and the number of components shown in Figure 5 is non-limiting.

[0039] In embodiments, the communication logic circuit 504 is configured to allow each main-controller 302 to communicate with the OTP memory 502. Advantageously, using a single OTP memory 502 for multiple systems 300 allows more efficient usage of memory space compared to the conventional design for test architecture, which included a dedicated OTP memory 402 for each system 300.

[0040] Figure 6 illustrates a partitioning of an embodiment OTP memory 502. As shown, OTP memory 502 is partitioned into n different partitions, where n is an integer greater than one-although OTP memory 502 is shown to include more than two partitions 602, the number of partitions 602 may be fewer or greater and non-limiting. In embodiments, each partition 602 includes an associated repair data (i.e., memory repair signature) for a respective system 300. In embodiments, each partition 602 of the OTP memory 502 is dedicated to a different system 300 in the multi-system 500.

[0041] Embodiments of this disclosure, thus, provide an OTP partitioning scheme and a method to address each partition 602 for each system 300 using an offset addition scheme when addressing the OTP memory 502. In embodiments, the repair data for the various systems 300 are sequentially stored within the OTP memory 502. Thus, the memory space wastage in the multi-system 500 is minimized by utilizing a single OTP memory 502 with sequential repair data loaded therein.

[0042] It should be understood that the OTP memory 502 and the size of the partitions 602 are decided based on, for example, the repair budget and user data per system 300. Thus, in some embodiments, the memory space of partition 602 is identical; in others, the partition memory space may be different.

[0043] For example, in an OTP memory 502, having four kilobytes of memory space and two partitions dedicated to two systems 300, the first partition 602 may have a one-kilobyte memory space, and the second partition 602 may have a three-kilobyte memory space. As another example, in an OTP memory 502, having four kilobytes of memory space and two partitions dedicated to two systems 300, each of the first partition 602 and the second partition 602 may have two kilobytes of memory space.

[0044] An associated offset is added to the address bus in embodiments when a main-controller 302 is configured to access the OTP memory 502. In embodiments, a null offset (i.e., zero offset) is added to the address bus when the main-controller 302 associated with the first partition 602 accesses the OTP memory 502. Further, the offset addition scheme is easily scalable to more than two controllers. In embodiments, the addition of the offset to the address bus occurs both in functional (i.e., boot) and test mode.

[0045] Figure 7 illustrates a block diagram of an embodiment design for system 700. The system 700 includes the OTP memory 502, an OTP interface circuit 702, an offset logic and selection circuit 704, a test control register (TCR) 716, a first main-controller 706a, a second main-controller 706b, a first repair controller 708a, a second repair controller 708b, an OTP sequencer circuit 710, functional logic 712, and an input/output (I/O) multiplexer 714, which may (or may not) be arranged as shown. System 700 can be used for functional and test mode operations. In embodiments, the TCR 716 is coupled to the TAP 304 such that the TAP 304 can control the operations of the TCR 716 during the test mode.

[0046] An optional OTP interface circuit 702, which is a logic circuit coupled to the OTP memory 502 and the offset logic and selection circuit 704, provides error correction code and other safety features in system 700. The OTP interface circuit 702 includes a status terminal, which is coupled to the I/O multiplexer 714, and configured to communicate status signals. In embodiments, the OTP interface circuit 702 is configured to generate an offset to the address signal based on which main-controller 706a-b communicates with the OTP memory 502.

[0047] The offset logic and selection circuit 704 is coupled to each main-controller 706a-b, the OTP interface circuit 702, the OTP sequencer circuit 710, and the TCR 716. The offset logic and selection circuit 704 provides for the offset addition.

[0048] Although Figure 7 includes a pair of main-controllers 706a-b, each coupled to a repair controller 708a-b, in embodiments, the system 700 may include more than two main-controllers, each being coupled to a respective repair controller. Thus, the number of main-controllers and repair controllers is not limiting.

[0049] Each main-controller 706a-b is coupled to a repair controller 708a-b and the offset logic and selection circuit 704. Further, each main-controller 706a-b may include one or more status terminals to exchange a respective status signal with the OTP sequencer circuit 710, functional logic 712 of the hardware accelerator 100, and the I/O multiplexer 714. In embodiments, the main-controllers 706a-b operate similarly to the main-controllers described hereinabove. For example, each main-controller 706a-b may be part of a respective system 300 as shown in Figure 5. In embodiments, each main-controller 706a-b is associated with a respective sub-set of the local memory 108 in the hardware accelerator 100.

[0050] The OTP sequencer circuit 710 is a logic circuit (e.g., operating as a memory repair control circuit during functional mode operation of the hardware accelerator 100) coupled to the main-controllers 706a-b, repair controllers 708a-b, and the offset logic and selection circuit 704. In embodiments, the OTP sequencer circuit 710 includes a reset terminal configured to receive a functional mode reset (POR_n) signal 802. In embodiments, the logic level of the functional mode reset signal 802 indicates whether the hardware accelerator 100 is operating in functional mode. For example, in response to the functional mode reset signal 802 having a logic level equal '0', the hardware accelerator is not yet booted up, and in response to the functional mode reset signal 802 having a logic level equal '1', the hardware accelerator is running and in functional mode.

[0051] The OTP sequencer circuit 710 is configured to generate grant signals during the functional mode operation of the hardware accelerator 100. The OTP sequencer circuit 710 generates a grant signal for each main-controller 706a-b to communicate with the OTP memory 502, during the functional mode. For example, in the embodiment of Figure 7, which includes a pair of main-controllers 706a-b, the OTP sequencer circuit 710 is configured to generate a first grant signal 804 for the first main-controller 706a and a second grant signal 808 for the second main-controller 706b, during the functional mode. In embodiments, when the first grant signal 804 is asserted, the second grant signal 808 is de-asserted. In embodiments, when the second grant signal 808 is asserted, the first grant signal 804 is de-asserted.

[0052] In embodiments, the OTP sequencer circuit 710 includes a dedicated terminal to communicate grant signals for each repair controller 708a-b (as shown). In

embodiments, a signal terminal of the OTP sequencer circuit 710 may be used to communicate grant signals to the repair controllers 708a-b (not shown). In other embodiments, other components, such as a multiplexer, may be combined to communicate the grant signals from the OTP sequencer circuit 710.

[0053] In embodiments, the first grant signal 804 is communicated from the OTP sequencer circuit 710 to the first repair controller 708a and the offset logic and selection circuit 704. In embodiments, the second grant signal 808 is communicated from the OTP sequencer circuit 710 to the second repair controller 708b and the offset logic and selection circuit 704.

[0054] The I/O multiplexer 714 is configured to provide a common selection interface between test status and other functional signals communicated from the main-controllers 706a-b and OTP interface circuit 702.

[0055] During functional mode (i.e., when the functional mode reset signal 802 is asserted), the logic level of the grant signals from the OTP sequencer circuit 710 indicates to the repair controllers 708a-b, which main-controller 706a-b is granted access to the OTP memory 502. A repair controller 708a-b instructs the main-controller 706a-b to (e.g., sequentially) load OTP repair data from the OTP memory 502 using the main-controller bus address 812 in response to being granted access to the OTP memory 502. The offset logic and selection circuit 704, which is coupled between the OTP memory 502 and the main-controller 706a-b, provides the appropriate offset to the main-controller bus address 812 to obtain the OTP bus address 814. The OTP bus address 814, thus, includes an offset indicating the partition 602, which the main-controller 706a-b is to access from the OTP memory 502.

[0056] The TCR 716 (e.g., operating as a memory repair control circuit during test mode operation of the hardware accelerator 100) is coupled to the offset logic and selection circuit 704. The TCR 716 is configured to generate grant signals and an access signal 1002 during the test mode operation. In embodiments, TCR 716 generates the access signal 1002 to indicate the operating in test mode. For example, in response to the access signal 1002 having a logic level equal '0', the system is not in test mode. In response to the access signal 1002 having a logic level equal to 'i', the system is in test mode.

[0057] The TCR 716 generates a corresponding grant signal for each main-controller 706a-b to communicate with the OTP memory 502, during test mode. For example, in the embodiment of Figure 7, which includes a pair of main-controllers 706a-b, the TCR 716 is configured to generate a first grant signal 1004 for the first main-controller 706a and a second grant signal 1006 for the second main-controller 706b, during test mode. In embodiments, when the first grant signal 1004 is asserted, the second grant signal 1006 is de-asserted. In embodiments, when the second grant signal 1006 is asserted, the first grant signal 1004 is de-asserted.

[0058] During test mode (i.e., when the access signal 1002 is asserted), the logic level of the grant signals from the TCR 716 indicates which main-controller 706a-b is granted access to the OTP memory 502. A system coupled instructs the main-controller 706a-b to (e.g., sequentially) load OTP repair data from the OTP memory 502 using the main-controller bus address 1008 in response to being granted access to the OTP memory 502. The offset logic and selection circuit 704, which is coupled between the OTP memory 502 and the main-controller 706a-b, provides the appropriate offset to the main-controller bus address 1008 to obtain the OTP bus address 1010. The OTP bus address 1010, thus, includes an offset indicating the partition 602, which the main-controller 706a-b is to access from the OTP memory 502.

[0059] Figures 8 and 9, respectively, illustrate an embodiment timing diagram 800 and a corresponding method 900 where system 700 operates in functional mode. The method starts at step 902, before time t_1, the functional mode reset (POR_n) signal 802 is de-asserted (e.g., logic level low). As the functional mode reset signal 802 is de-asserted, the system has not yet booted up. At step 904, at time t_1, the functional mode reset signal 802 is asserted (e.g., logic level high), and hardware accelerator 100 enters boot mode (i.e., functional mode).

[0060] At step 906, at time t_2, OTP sequencer circuit 710 asserts (e.g., logic level high) a first grant signal 804 for the first main-controller 706a and the first repair controller 708a. In response to receiving the first grant signal 804 from the OTP sequencer circuit 710, the first repair controller 708a instructs the first main-controller 706a to load OTP repair data. As the associated partition 602 for the first main-controller 706a begins at address '0', the main-controller bus address 812 is identical to the OTP bus address 814, with a null offset. At step 908, between time t_3 and time t_4, the first main-controller 706a loads OTP repair data from the first partition 602 of the OTP memory 502 based on the OTP bus address 814 and de-asserts (e.g., logic level low) a first status signal 806.

[0061] At step 910, after loading the OTP repair data from the first partition 602 of OTP memory 502, by the first main-controller 706a at time t_4, the first status signal 806 is asserted (e.g., logic level high).

[0062] In response, at step 912 and at time t_5, OTP sequencer circuit 710 asserts (e.g., logic level high) a second grant signal 808 for the second main-controller 706b and de-asserts (e.g., logic level high) the first grant signal 804 for the first main-controller 706a. Further, repair controller 708 instructs the second main-controller 706b to load OTP repair data. However, during this sequence, the offset logic and selection circuit 704 adds an offset to the main-controller bus address 812 to generate the OTP bus address 814 from which the OTP repair data begins loading (i.e., second partition 602). The OTP bus address 814, thus, points to the second partition 602 in the OTP memory 502.

[0063] At step 914, between time t_6 and time t_7, the second main-controller 706b fetches OTP repair data

from the second partition 602 of the OTP memory 502 based on the OTP bus address 814 and de-asserts (e.g., logic level low) a second status signal 810.

[0064] At step 916, after completion of the loading of the OTP repair data from the second partition 602 of OTP memory 502, by the second main-controller 706b at time ts, the second status signal 810 is asserted (e.g., logic level high). At step 918, OTP sequencer circuit 710 de-asserts (e.g., logic level high) the second grant signal 808 for the second main-controller 706b.

[0065] Figures 10 and 11, respectively, illustrate an embodiment timing diagram 1000 and a corresponding method 1100 where the system 700 operates in test mode. The method starts at step 1102, before time t_1, the access signal 1002 is de-asserted (e.g., logic level low). As the access signal 1002 is de-asserted, the hardware accelerator 100 is not yet in test mode. At step 1104, at time t_1, the access signal 1002 is asserted (e.g., logic level high), and hardware accelerator 100 enters test mode.

[0066] At step 1106, at time t_2, TCR 716 asserts (e.g., logic level high) a first grant signal 1004 for the first main-controller 706a, which instructs the first main-controller 706a to load OTP repair data. As the associated partition 602 for the first main-controller 706a begins at address '0', the main-controller bus address 812 is identical to the OTP bus address 814, with a null offset. At step 1108, between time t_2 and time t_3, the first main-controller 706a loads OTP repair data from the first partition 602 of the OTP memory 502 based on the OTP bus address 814.

[0067] At step 1110, after loading the OTP repair data from the first partition 602 of OTP memory 502, by the first main-controller 706a at time t_3, TCR 716 de-asserts (e.g., logic level low) the first grant signal 1004. In embodiments, the timing is predetermined as there is typically no feedback during test mode, and manufacturing tests-the test pattern times are fixed and generally independent from a feedback signal.

[0068] At step 1112, at time t_4, TCR 716 asserts (e.g., logic level high) a second grant signal 1006 for the second main-controller 706b, instructs the second main-controller 706b to load OTP repair data. However, during this sequence, the offset logic and selection circuit 704 adds an offset to the main-controller bus address 812 to generate the OTP bus address 814 from which the OTP repair data begins loading (i.e., second partition 602). The OTP bus address 814, thus, points to the second partition 602 in the OTP memory 502.

[0069] At step 1114, between time t_4 and time t_3, the second main-controller 706b fetches OTP repair data from the second partition 602 of the OTP memory 502 based on the OTP bus address 814.

[0070] At step 1116, after completion of the loading of the OTP repair data from the second partition 602 of OTP memory 502, by the second main-controller 706b at time t_5, the second grant signal is de-asserted (e.g., logic level low).

[0071] It is noted that all steps outlined in the flow charts of methods 900 and 1100 are not necessarily required and can be optional. Further, changes to the arrangement of the steps, removal of one or more steps and path connections, and addition of steps and path connections are similarly contemplated.

[0072] A first aspect relates to a system for testing and repairing of local memory in a hardware accelerator. The system includes a one-time programmable (OTP) memory, an OTP sequencer circuit, a first main-controller, a second main-controller, and an offset addition and selection logic circuit. The OTP memory is configured to sequentially store first repair data in a first partition and second repair data in a second partition for, respectively, a first subset of the local memory and a second subset of the local memory. The OTP sequencer circuit is configured to assert a first grant signal. The first main-controller is coupled to the OTP sequencer circuit and the first subset of the local memory. The first main-controller configured to first load the first repair data for the first subset of the local memory from the first partition, and communicate a first status signal to the OTP sequencer circuit after completion of the first loading indicating a completion of the first loading. The OTP sequencer circuit is further configured to de-assert the first grant signal in response to the OTP sequencer circuit receiving the first status signal and assert a second grant signal in response to the de-asserting of the first grant signal. The offset addition and selection logic circuit is configured to add an offset to an address used by a second main-controller to access the second partition. The second main-controller configured to second load the second repair data for the second subset of the local memory associated with the second main-controller from the second partition using the address to access the second partition, and communicate a second status signal to the OTP sequencer circuit after completion of the second loading indicating a completion of the second loading. The OTP sequencer circuit is further configured to de-assert the second grant signal in response to the OTP sequencer circuit receiving the second status signal.

[0073] In a first implementation form of the system according to the first aspect as such, the OTP sequencer circuit is configured to generate the first grant signal in response to the OTP sequencer circuit receiving a functional mode reset signal.

[0074] In a second implementation form of the system according to the first aspect as such or any preceding implementation form of the first aspect, the system further includes a first repair controller coupled to the first main-controller and the OTP sequencer circuit. The OTP sequencer circuit is configured to instruct the first main-controller to load the first repair data by the first repair controller in response to receiving the first grant signal.

[0075] In a third implementation form of the system according to the first aspect as such or any preceding implementation form of the first aspect, the system further includes a second repair controller coupled to the second main-controller and the OTP sequencer circuit. The OTP

sequencer circuit is configured to instruct the second main-controller to load the second repair data by the second repair controller in response to receiving the second grant signal.

[0076] In a fourth implementation form of the system according to the first aspect as such or any preceding implementation form of the first aspect, the hardware accelerator includes a Graphics Processing Unit (GPU), a Field Programmable Gate Array (FPGA), an Application-Specific Integrated Circuit (ASIC), or a combination thereof.

[0077] In a fifth implementation form of the system according to the first aspect as such or any preceding implementation form of the first aspect, the hardware accelerator is used for artificial intelligence or machine learning.

[0078] In a sixth implementation form of the system according to the first aspect as such or any preceding implementation form of the first aspect, the second repair data is sequentially stored after the first repair data in the OTP memory such that the offset added to the address points to a memory space immediately after the first repair data.

[0079] A second aspect relates to a method for testing and repairing of local memory in a hardware accelerator from a one-time programmable memory (OTP). The method includes asserting, by a one-time programable (OTP) sequencer circuit, a first grant signal; a first loading, by a first main-controller, of a first repair data for a first sub-set of the local memory associated with the first main-controller from a first partition of the OTP memory; communicating, by the first main-controller, a first status signal to the OTP sequencer circuit after completion of the first loading indicating a completion of the first loading; de-asserting, by the OTP sequencer circuit, the first grant signal in response to the OTP sequencer circuit receiving the first status signal; asserting, by the OTP sequencer circuit, a second grant signal in response to the de-asserting of the first grant signal; adding, by an offset addition and selection logic circuit, an offset to an address used by a second main-controller to access a second partition of the OTP memory comprising a second repair data, wherein the first repair data and the second repair data are sequentially loaded in the OTP memory; a second loading, by the second main-controller, of the second repair data for a second sub-set of the local memory associated with the second main-controller from the second partition of the OTP memory using the address to access the second partition; communicating, by the second main-controller, a second status signal to the OTP sequencer circuit after completion of the second loading indicating a completion of the second loading; and de-asserting, by the OTP sequencer circuit, the second grant signal in response to the OTP sequencer circuit receiving the second status signal.

[0080] In a first implementation form of the method according to the second aspect as such, the first grant signal is generated in response to the OTP sequencer circuit receiving a functional mode reset signal.

[0081] In a second implementation form of the method according to the second aspect as such or any preceding implementation form of the second aspect, the method further includes instructing the first main-controller to load the first repair data by a first repair controller in response to receiving the first grant signal. The first repair controller is coupled to the first main-controller and the OTP sequencer circuit.

[0082] In a third implementation form of the method according to the second aspect as such or any preceding implementation form of the second aspect, the method further includes instructing the second main-controller to load the second repair data by a second repair controller in response to receiving the second grant signal. The second repair controller is coupled to the second main-controller and the OTP sequencer circuit.

[0083] In a fourth implementation form of the method according to the second aspect as such or any preceding implementation form of the second aspect, the hardware accelerator includes a Graphics Processing Unit (GPU), a Field Programmable Gate Array (FPGA), an Application-Specific Integrated Circuit (ASIC), or a combination thereof.

[0084] In a fifth implementation form of the method according to the second aspect as such or any preceding implementation form of the second aspect, the hardware accelerator is used for artificial intelligence or machine learning.

[0085] In a sixth implementation form of the method according to the second aspect as such or any preceding implementation form of the second aspect, the method further includes storing the second repair data sequentially after the first repair data in the OTP memory such that the offset added to the address points to a memory space immediately after the first repair data.

[0086] A third aspect relates to a system for testing and repairing of local memory in a hardware accelerator. The system includes a one-time programmable (OTP) memory, a test control register (TCR), a first main-controller, a second main-controller, and an offset addition and selection logic circuit. The OTP memory is configured to sequentially store first repair data in a first partition and second repair data in a second partition for, respectively, a first subset of the local memory and a second subset of the local memory. The TCR is configured to assert a first grant signal. The first main-controller is coupled to the first subset of the local memory, the first main-controller configured to first load the first repair data for the first subset of the local memory from the first partition in response to the first grant signal being asserted. The offset addition and selection logic circuit coupled to the TCR and the first main-controller. The offset addition and selection logic circuit configured to add an offset to an address used by a second main-controller to access the second partition. The TCR is further configured to de-assert the first grant signal and assert a second grant signal in response to the de-asserting of the first grant

signal. The second main-controller configured to second load the second repair data for the second subset of the local memory associated with the second main-controller from the second partition using the address to access the second partition in response to the second grant signal being asserted, wherein the TCR is further configured to de-assert the second grant signal.

[0087] In a first implementation form of the system according to the third aspect as such, the TCR is further configured to assert an access signal to operate the hardware accelerator in test mode.

[0088] In a second implementation form of the system according to the third aspect as such or any preceding implementation form of the third aspect, the hardware accelerator includes a Graphics Processing Unit (GPU), a Field Programmable Gate Array (FPGA), an Application-Specific Integrated Circuit (ASIC), or a combination thereof.

[0089] In a third implementation form of the system according to the third aspect as such or any preceding implementation form of the third aspect, the hardware accelerator is used for artificial intelligence or machine learning.

[0090] In a fourth implementation form of the system according to the third aspect as such or any preceding implementation form of the third aspect, the second repair data is sequentially stored after the first repair data in the OTP memory such that the offset added to the address points to a memory space immediately after the first repair data.

[0091] In a fifth implementation form of the system according to the third aspect as such or any preceding implementation form of the third aspect, the system includes a test access point couplable to the TCR, the TAP being configured to control operations of the TCR during a test mode operation of the hardware accelerator.

[0092] Although the description has been described in detail, it should be understood that various changes, substitutions, and alterations may be made without departing from the spirit and scope of this disclosure as defined by the appended claims. The same elements are designated with the same reference numbers in the various figures. Moreover, the scope of the disclosure is not intended to be limited to the particular embodiments described herein, as one of ordinary skill in the art will readily appreciate from this disclosure that processes, machines, manufacture, compositions of matter, means, methods, or steps, presently existing or later to be developed, may perform substantially the same function or achieve substantially the same result as the corresponding embodiments described herein. Accordingly, the appended claims are intended to include within their scope such processes, machines, manufacture, compositions of matter, means, methods, or steps.

[0093] The specification and drawings are, accordingly, to be regarded simply as an illustration of the disclosure as defined by the appended claims, and are contemplated to cover any and all modifications, variations, combinations, or equivalents that fall within the scope of the present disclosure.

Claims

1. A hardware accelerator (100) comprising a system (500, 700) for testing and repairing local memory (108) of the hardware accelerator (100), the system (500, 700) comprising:

   a one-time programmable, OTP, memory (502) configured to sequentially store first repair data in a first partition (602) and second repair data in a second partition (602) for, respectively, a first subset of the local memory (108) and a second subset of the local memory (108);
   a memory repair control circuit (710, 716) configured to sequentially assert a first grant signal (804; 1004) and a second grant signal (808; 1006);
   a first main-controller (706a) coupled to the first subset of the local memory (108), the first main-controller (706a) configured to first load the first repair data for the first subset of the local memory (108) from the first partition (602) in response to the first grant signal (804; 1004) being asserted,
   an offset addition and selection logic circuit (704) configured to add an offset to an address used by a second main-controller (706b) to access the second partition (602); and
   a second main-controller (706b) coupled to the second subset of the local memory (108), the second main-controller (706b) configured to second load the second repair data for the second subset of the local memory (108) from the second partition (602) in response to the second grant signal (808; 1006) being asserted, using the address to access the second partition (602).

2. The hardware accelerator (100) of claim 1, wherein:

   said memory repair control circuit comprises an OTP sequencer circuit (710);
   said first main-controller (706a) and said second main-controller (706b) are coupled to the OTP sequencer circuit (710);
   the first main-controller (706a) is further configured to communicate a first status signal (806) to the OTP sequencer circuit (710) after completion of the first loading indicating a completion of the first loading of the first repair data;
   the OTP sequencer circuit (710) is further configured to de-assert the first grant signal (804) in response to the OTP sequencer circuit (710) receiving the first status signal (806) and assert

the second grant signal (808) in response to the de-asserting of the first grant signal (804);
the second main-controller (706b) is further configured to communicate a second status signal (810) to the OTP sequencer circuit (710) after completion of the second loading indicating a completion of the second loading of the second repair data; and
the OTP sequencer circuit (710) is further configured to de-assert the second grant signal (808) in response to the OTP sequencer circuit (710) receiving the second status signal (810).

3. The hardware accelerator (100) of claim 2, wherein the OTP sequencer circuit (710) is configured to assert the first grant signal (804) in response to the OTP sequencer circuit (710) receiving a functional mode reset signal (802, POR_n).

4. The hardware accelerator (100) of claim 2 or claim 3, further comprising:

a first repair controller (708a) coupled to the first main-controller (706a) and the OTP sequencer circuit (710), wherein the OTP sequencer circuit (710) is configured to instruct the first main-controller (706a) to load the first repair data by the first repair controller (708a) in response to receiving the first grant signal (804); and/or
a second repair controller (708b) coupled to the second main-controller (706b) and the OTP sequencer circuit (710), wherein the OTP sequencer circuit (710) is configured to instruct the second main-controller (706b) to load the second repair data by the second repair controller (708b) in response to receiving the second grant signal (808).

5. The hardware accelerator (100) of claim 1, wherein:

the memory repair control circuit comprises a test control register, TCR (716);
the offset addition and selection logic circuit (704) is coupled to the TCR (716);
the TCR (716) is further configured to de-assert the first grant signal (1004), assert the second grant signal (1006) in response to the de-asserting of the first grant signal (1004), and de-assert the second grant signal (1006).

6. The hardware accelerator (100) of claim 5, wherein the TCR (716) is further configured to assert an access signal (1002) to operate the hardware accelerator (100) in test mode.

7. The hardware accelerator (100) of claim 5 or claim 6, further comprising a test access point, TAP (304) couplable to the TCR (716), the TAP (304) configured to control operations of the TCR (716) during a test mode operation of the hardware accelerator (100).

8. The hardware accelerator (100) of any of the previous claims, wherein the second repair data is sequentially stored after the first repair data in the OTP memory (502) such that the offset added to the address points to a memory space immediately after the first repair data.

9. A method (900; 1100) for testing and repairing local memory (108) in a hardware accelerator (100) from a one-time programmable, OTP, memory (502), the method comprising:

asserting (906; 1106), by a memory repair control circuit (710, 716), a first grant signal (804; 1004);
first loading (908; 1108), by a first main-controller (706a) coupled to a first subset of the local memory (108), first repair data for the first subset of the local memory (108) from a first partition (602) of the OTP memory (502) in response to the first grant signal (804; 1004) being asserted;
asserting (912; 1112), by the memory repair control circuit (710, 716), a second grant signal (808; 1006);
adding, by an offset addition and selection logic circuit (704), an offset to an address used by a second main-controller (706b) coupled to a second subset of the local memory (108) to access a second partition (602) of the OTP memory (502) comprising second repair data, wherein the first repair data and the second repair data are sequentially stored in the OTP memory (502); and
second loading (914; 1114), by the second main-controller (706b), the second repair data for the second subset of the local memory (108) from the second partition (602) of the OTP memory (502) using the address to access the second partition (602) in response to the second grant signal (808; 1006) being asserted.

10. The method (900) of claim 9, wherein said memory repair control circuit (710, 716) comprises an OTP sequencer circuit (710), the method (900) further comprising:

communicating (910), by the first main-controller (706a), a first status signal (806) to the OTP sequencer circuit (710) after completion of the first loading indicating a completion of the first loading of the first repair data;
de-asserting (912), by the OTP sequencer circuit (710), the first grant signal (804) in response to the OTP sequencer circuit (710) receiving the first status signal (806);

asserting (912), by the OTP sequencer circuit (710), the second grant signal (808) in response to the de-asserting of the first grant signal (804);
communicating (916), by the second main-controller (706b), a second status signal (810) to the OTP sequencer circuit (710) after completion of the second loading indicating a completion of the second loading of the second repair data; and
de-asserting (918), by the OTP sequencer circuit (710), the second grant signal (808) in response to the OTP sequencer circuit (710) receiving the second status signal (810).

11. The method (900) of claim 10, wherein the first grant signal (804) is asserted (906) in response to the OTP sequencer circuit (710) receiving (904) a functional mode reset signal (802, POR_n).

12. The method (900) of claim 10 or claim 11, further comprising:

instructing the first main-controller (706a) to load the first repair data by a first repair controller (708a) in response to receiving the first grant signal (804), wherein the first repair controller (708a) is coupled to the first main-controller (706a) and the OTP sequencer circuit (710); and/or
instructing the second main-controller (706b) to load the second repair data by a second repair controller (708b) in response to receiving the second grant signal (808), wherein the second repair controller (708b) is coupled to the second main-controller (706b) and the OTP sequencer circuit (710).

13. The method (1100) of claim 9, wherein said memory repair control circuit (710, 716) comprises a test control register, TCR (716), the method (1100) comprising:

de-asserting (1110), by the TCR (716), the first grant signal (1004);
asserting (1112), by the TCR (716), the second grant signal (1006) in response to the de-asserting of the first grant signal (1004); and
de-asserting (1116), by the TCR (716), the second grant signal (1006).

14. The method (1100) of claim 13, further comprising asserting (1104), by the TCR (716), an access signal (1002) to operate the hardware accelerator (100) in test mode.

15. The method (1100) of claim 13 or claim 14, wherein the hardware accelerator (100) further comprises a test access point, TAP (304), couplable to the TCR (716), the method (1100) further comprising controlling operations of the TCR (716) by the TAP (304) during a test mode operation of the hardware accelerator (100).

16. The method of any of claims 9 to 15, further comprising storing the second repair data sequentially after the first repair data in the OTP memory (502) such that the offset added to the address points to a memory space immediately after the first repair data.

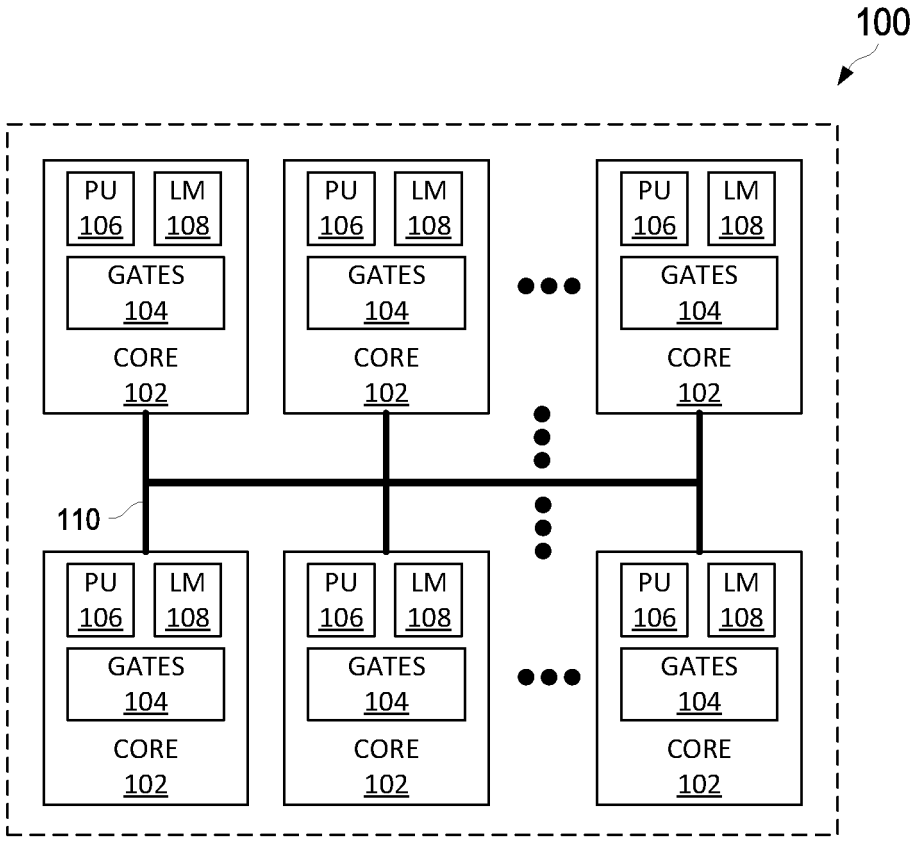
100
PU
106
LM
108
GATES
104
CORE
102
110

FIG. 1

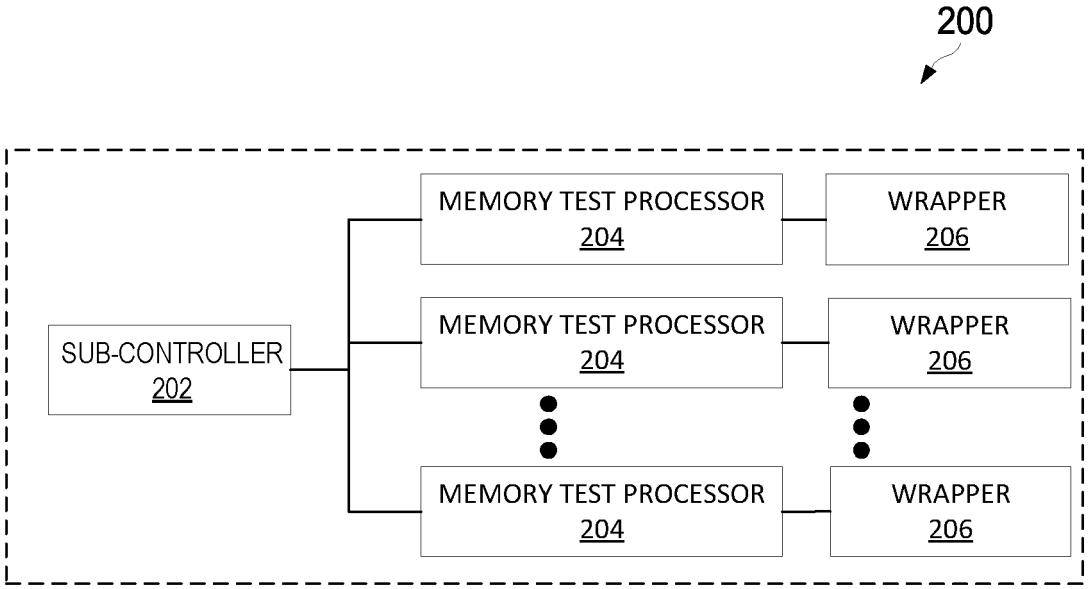
200
SUB-CONTROLLER
202
MEMORY TEST PROCESSOR
204
WRAPPER
206

FIG. 2

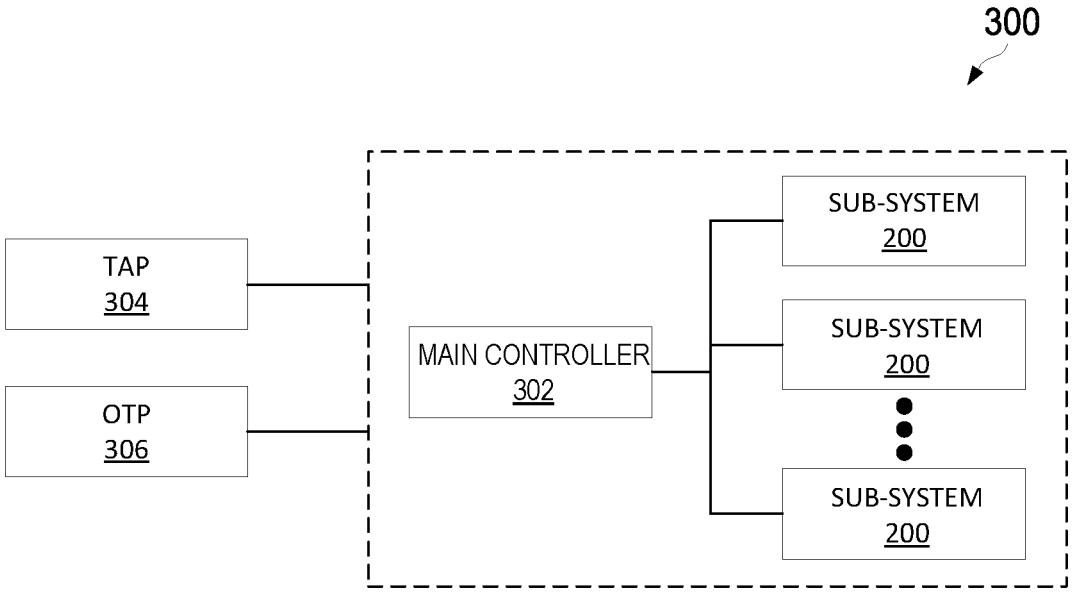
300
TAP
304
OTP
306
MAIN CONTROLLER
302
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200

FIG. 3

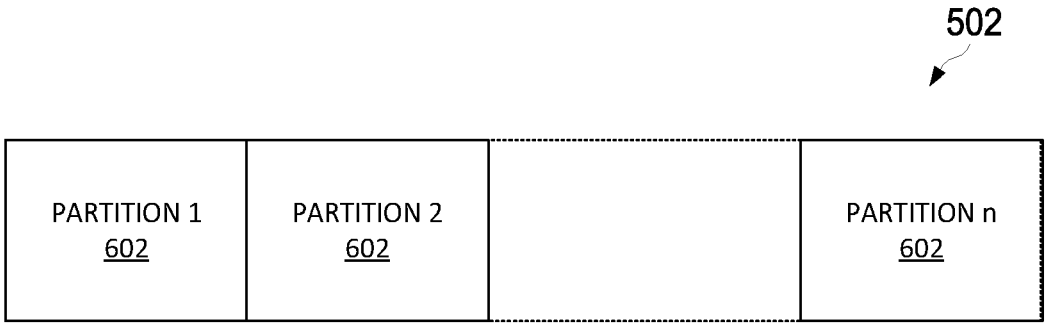
502
PARTITION 1
602
PARTITION 2
602
PARTITION n
602

FIG. 6

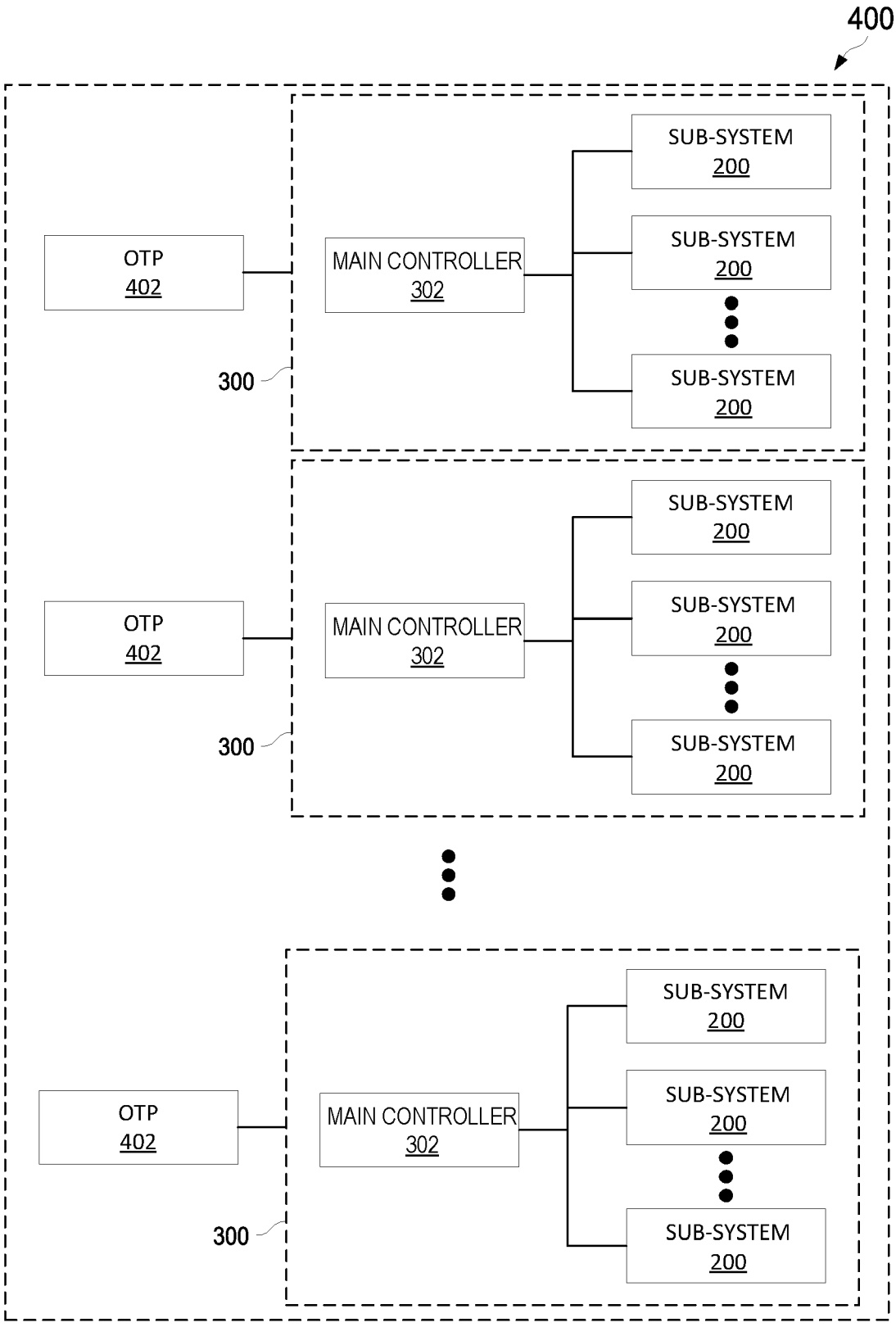
400
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200
OTP
402
MAIN CONTROLLER
302
300
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200
OTP
402
MAIN CONTROLLER
302
300
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200
OTP
402
MAIN CONTROLLER
302
300

FIG. 4

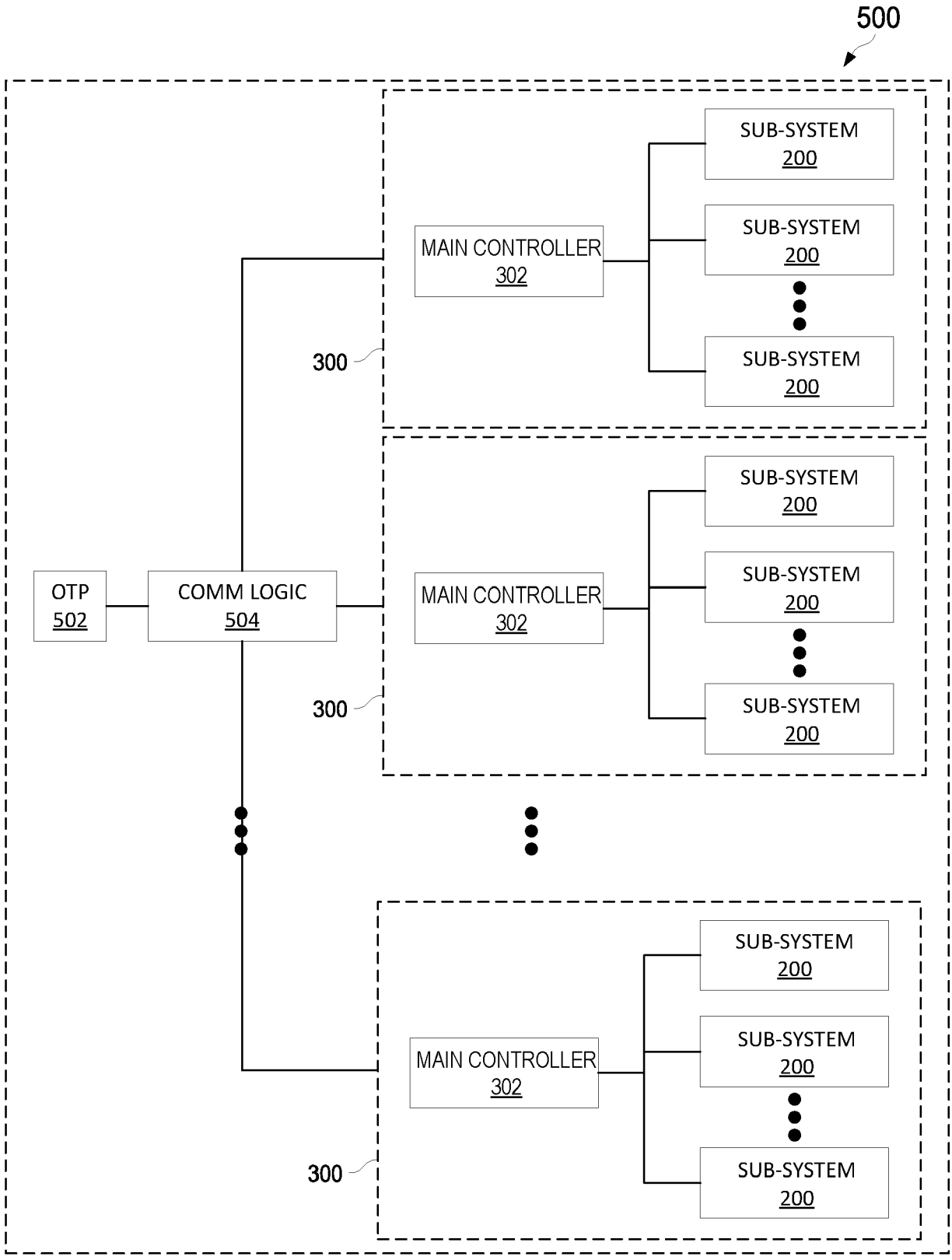
500
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200
MAIN CONTROLLER
302
300
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200
MAIN CONTROLLER
302
300
OTP
502
COMM LOGIC
504
SUB-SYSTEM
200
SUB-SYSTEM
200
SUB-SYSTEM
200
MAIN CONTROLLER
302
300

FIG. 5

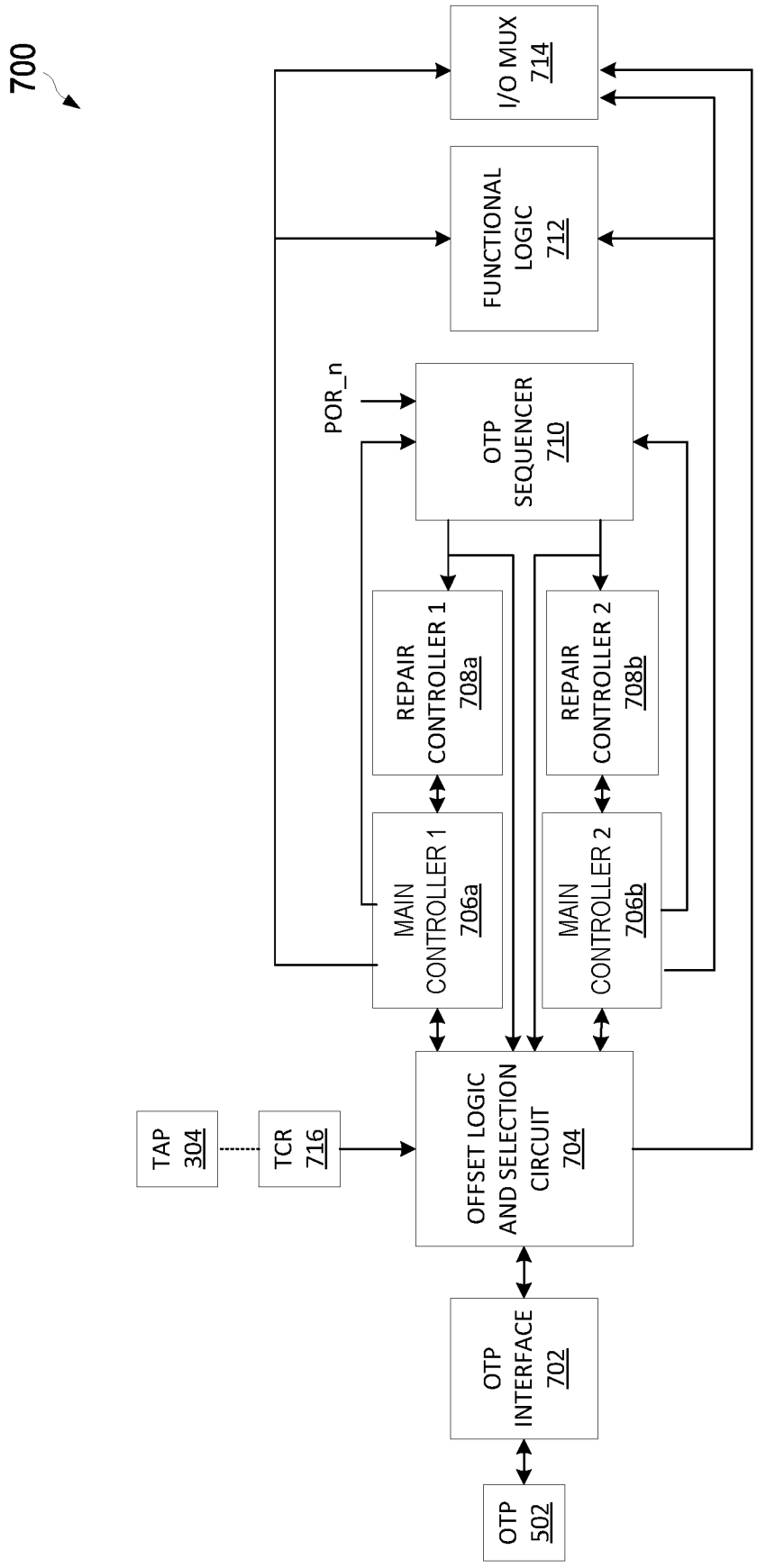
700
I/O MUX 714
FUNCTIONAL LOGIC 712
POR_n
OTP SEQUENCER 710
REPAIR CONTROLLER 1 708a
REPAIR CONTROLLER 2 708b
MAIN CONTROLLER 1 706a
MAIN CONTROLLER 2 706b
TAP 304
TCR 716
OFFSET LOGIC AND SELECTION CIRCUIT 704
OTP INTERFACE 702
OTP 502

FIG. 7

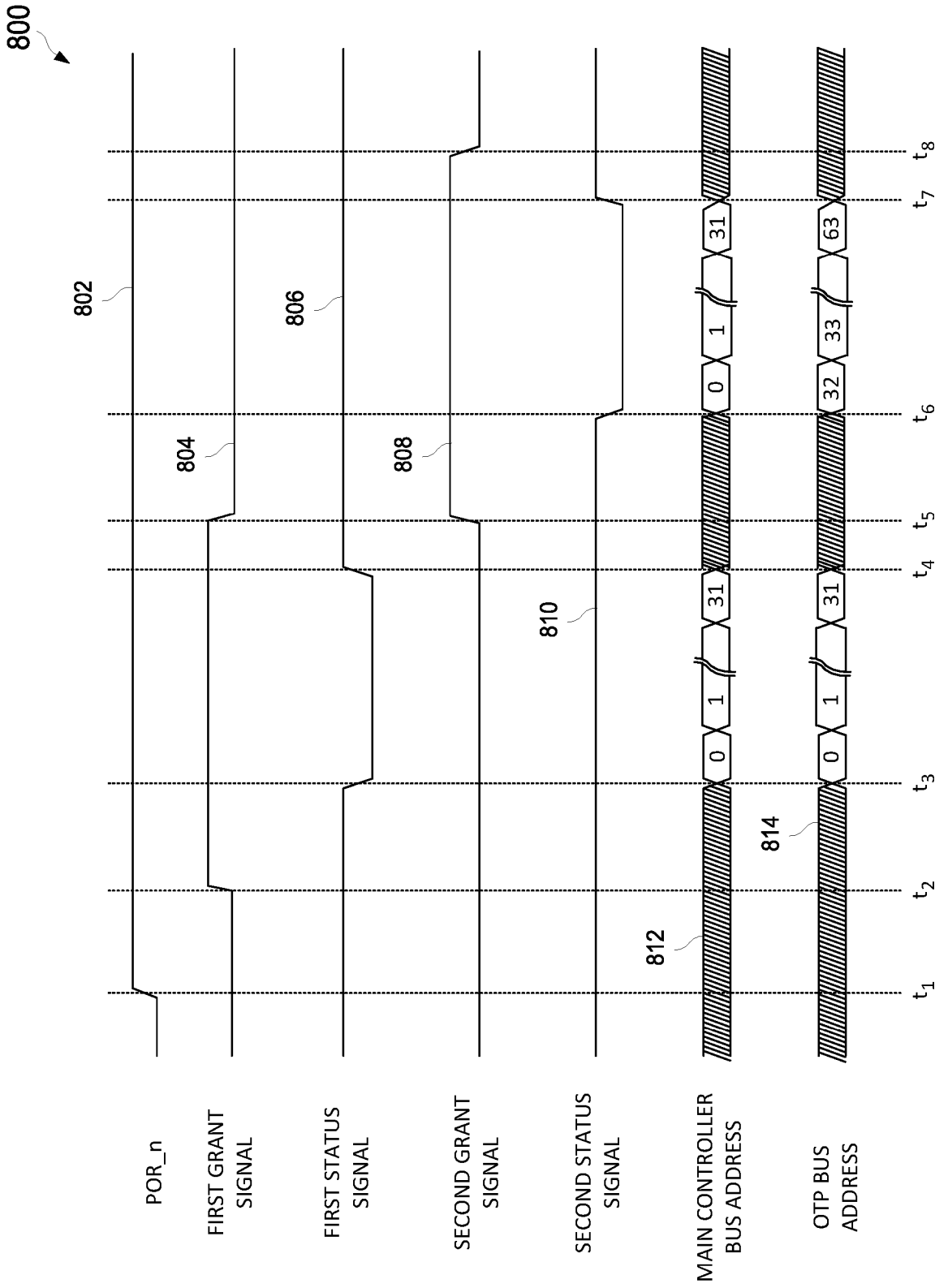
800
802
804
806
808
810
812
814
POR_n
FIRST GRANT SIGNAL
FIRST STATUS SIGNAL
SECOND GRANT SIGNAL
SECOND STATUS SIGNAL
MAIN CONTROLLER BUS ADDRESS
OTP BUS ADDRESS
0
1
31
32
33
63
t1
t2
t3
t4
t5
t6
t7
t8

FIG. 8

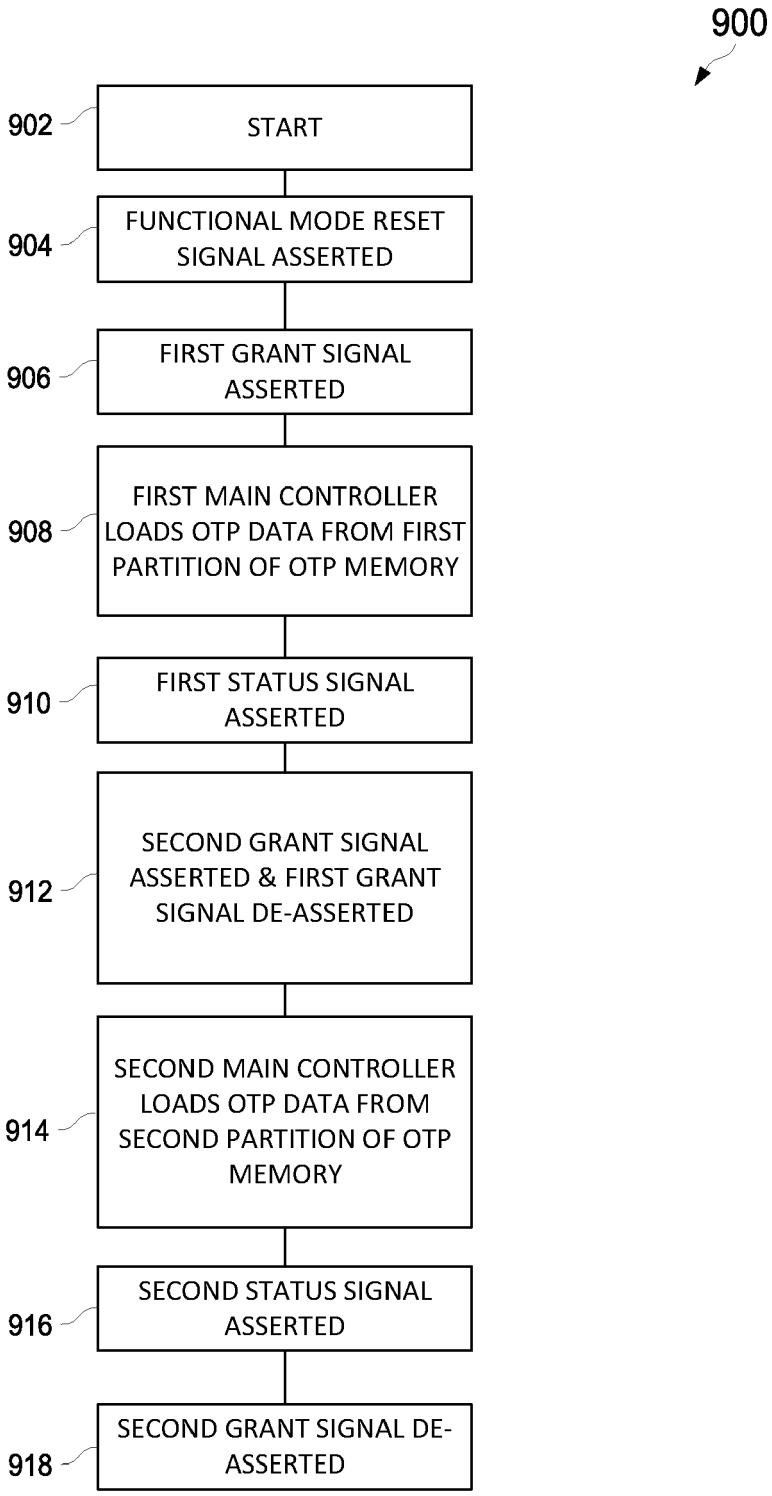
900
902
START
904
FUNCTIONAL MODE RESET SIGNAL ASSERTED
906
FIRST GRANT SIGNAL ASSERTED
908
FIRST MAIN CONTROLLER LOADS OTP DATA FROM FIRST PARTITION OF OTP MEMORY
910
FIRST STATUS SIGNAL ASSERTED
912
SECOND GRANT SIGNAL ASSERTED & FIRST GRANT SIGNAL DE-ASSERTED
914
SECOND MAIN CONTROLLER LOADS OTP DATA FROM SECOND PARTITION OF OTP MEMORY
916
SECOND STATUS SIGNAL ASSERTED
918
SECOND GRANT SIGNAL DE-ASSERTED

FIG. 9

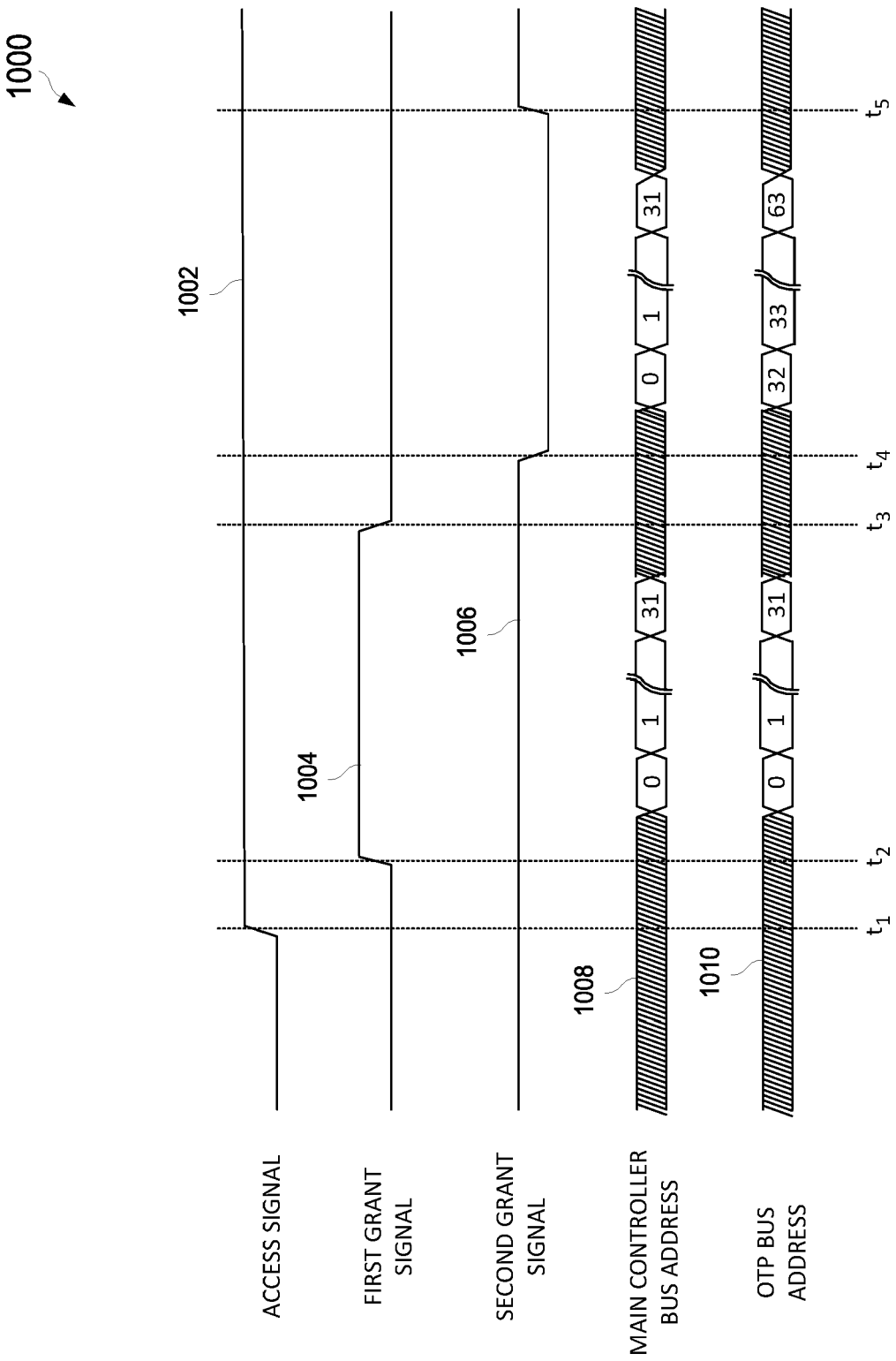
1000
1002
1004
1006
1008
1010
ACCESS SIGNAL
FIRST GRANT SIGNAL
SECOND GRANT SIGNAL
MAIN CONTROLLER BUS ADDRESS
OTP BUS ADDRESS
0
1
31
32
33
63
t_1
t_2
t_3
t_4
t_5

FIG. 10

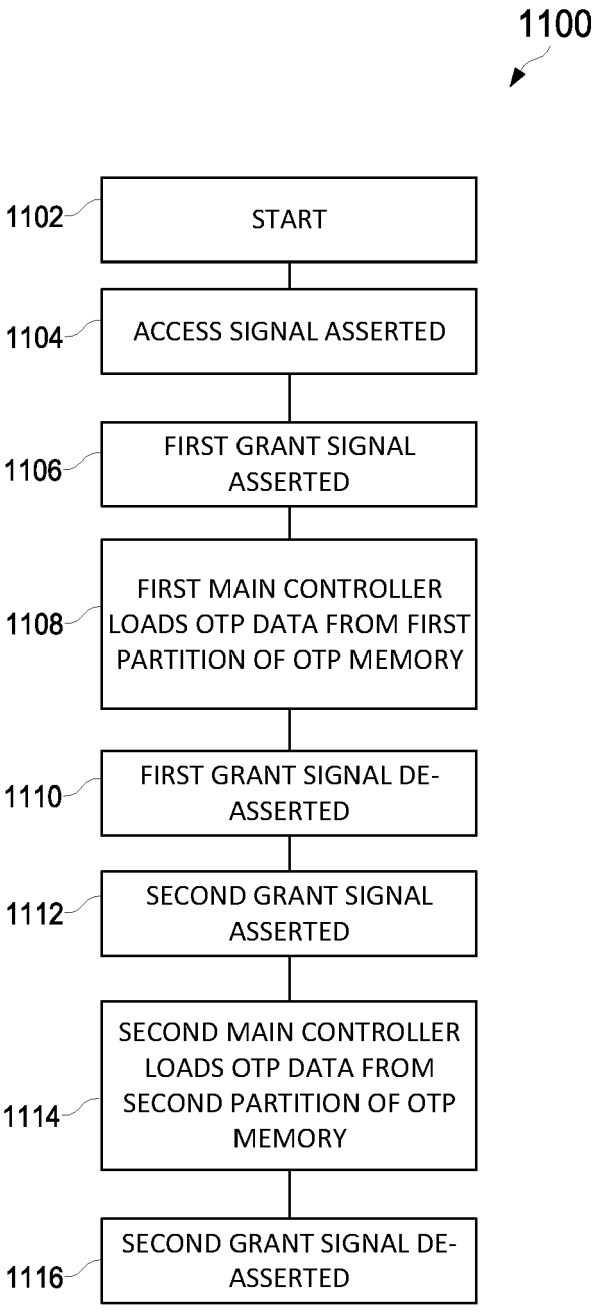
1100
1102
START
1104
ACCESS SIGNAL ASSERTED
1106
FIRST GRANT SIGNAL ASSERTED
1108
FIRST MAIN CONTROLLER LOADS OTP DATA FROM FIRST PARTITION OF OTP MEMORY
1110
FIRST GRANT SIGNAL DE-ASSERTED
1112
SECOND GRANT SIGNAL ASSERTED
1114
SECOND MAIN CONTROLLER LOADS OTP DATA FROM SECOND PARTITION OF OTP MEMORY
1116
SECOND GRANT SIGNAL DE-ASSERTED

FIG. 11

Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 24 17 8167

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X A	US 2011/029813 A1 (GUNDERSON ROSALEE [US] ET AL) 3 February 2011 (2011-02-03) * figures 1-3,6 * * paragraphs [0009], [0032] - [0056], [0067] - [0071] * -----	1,5-9, 13-16 2-4, 10-12	INV. G11C29/00 G11C5/02 G11C29/44 ADD. G11C7/10 G11C17/18 G11C11/54

TECHNICAL FIELDS SEARCHED (IPC)
G11C

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	5 July 2024	Vlachogiannakis, G

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 24 17 8167

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

05-07-2024

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2011029813 A1	03-02-2011	NONE	

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82